Design of Low Pass FIR Filter Using Artificial Neural Network

Harpreet Kaur and Balwinder Dhaliwal

***Abstract*—In signal processing, there are many instances in which an input signal to a system contains extra unnecessary content or additional noise which can degrade the quality of the desired portion. In such cases we may remove or filter out the useless samples. For example, in the case of the telephone system, there is no reason to transmit very high frequencies since most speech falls within the band of 400 to 3,400 Hz. Therefore, in this case, all frequencies above and below that band are filtered out. The frequency band between 400 and 3,400 Hz, which isn't filtered out, is known as the pass band, and the frequency band that is blocked out is known as the stop band.[1] Finite Impulse Response, filters are one of the primary types of filters used in Digital Signal Processing. For the design of Low pass FIR filters complex calculations are required. Mathematically, by substituting the values of Pass band, transition width, pass band ripple, stop band attenuation, sampling frequency in any of the methods from window method, frequency sampling method or optimal method we can get the values of filter coefficients h(n)[2].In this paper, Kaiser Window method has been chosen preferably because of the presence of ripple factor (β). Considering Low pass Filter design, the range of values for the parameters required are calculated. A data sheet through programming is performed on the platform of Matlab. For 30 different range of parameters, the values of h(n) i.e. coefficients of FIR filter, named desired result have been calculated .Artificial Neural Network is a highly simplified model of the structure of the biological neural network. It consists of interconnected processing units. In this thesis, ANN model has been designed which is used to design the low pass FIR which in the specified range of parameter which has been used to train the neural network. Basically, ANN can be trained by many methods like Feed forward neural network, Feedback neural network. But in this is paper the feed forward neural network has been chosen to train the network. Here radial basis function in neural networks is used for the training of the neural network.**

***Index Terms*—Artificial neural network, digital filter, signal processing.**

I. Artificial Neural Network

Artificial Neural Networks (ANNs) are computing systems made up of a number of simple, highly interconnected processing elements, which processes information by their dynamic state response to external inputs.

Garrett has given an interesting engineering definition of the ANN as: "a computational mechanism able to acquire, represent and computer mapping from one multivariate space of information to another, given a set of data representing that mapping". One of the distinct characteristics of the ANN is its ability to learn and generalize from experience and examples and to adapt to changing situations. ANNs are able to map causal models (i.e. mapping from cause to effect for estimation and prediction) and inverse mapping (i.e. mapping from effect to possible cause) [3].

ANNs are models of real world problems which map from a set of given patterns (input patterns)to an associated set of known values (target output).In simple terms a neural network tries to imitate some of the learning activities of the human brain. ANNs are much simpler than the human brain. They comprise fewer components and operate in a manner that is greatly abstract. The training process in the MLP network involves presenting a set of examples (input patterns) with known outputs (target output). The system adjusts the weights of the internal connections to minimize errors between the network output and target output. After the neural network is satisfactorily trained and tested, it is able to generalize rules and will be able to respond to unseen input data to predict required output, within the domain covered by the training examples.

Artificial Neural Networks are applied to perform many different tasks, such as filtering, Identification, control, prediction, etc. Among the most common tasks of ANNs are:

Classification: Where the input to the network is an encoded description of an object to be recognized, and the output is generally a binary identifier of the class of the presented object. The ANN can learn linear or non-linear class boundaries from examples.

Function approximation: Where the ANN performs mapping from the input space to the output space. Vectors in both spaces generally use real number components. Neural networks are then function approximation tools that replace lengthy procedural algorithms.(Rafiq et al.2001)

Advantages:

- A neural network can perform tasks that a linear program can not.
- When an element of the neural network fails, it can continue without any problem by their parallel nature.
- A neural network learns and does not need to be reprogrammed.
- It can be implemented in any application.

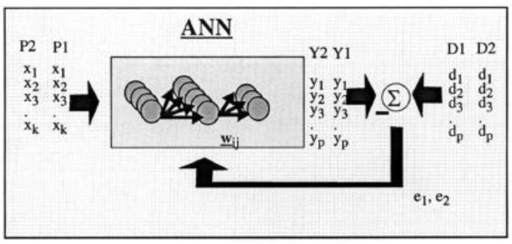
Another aspect of the artificial neural networks is that there are different architectures, which consequently requires different types of algorithms, but despite to be an apparently complex system, a neural network is relatively simple.

Artificial neural networks (ANN) are among the newest signal-processing technologies in the engineer's toolbox. The field is highly interdisciplinary, but our approach will restrict the view to the engineering perspective. In engineering, neural networks serve two important functions:

Manuscript received August 15, 2012; revised October 12, 2012.
Harpreet Kaur is with BBSBEC, Fatehgarh Sahib (e-mail: mavi_preeti@yahoo.co.in, harpreet.mavi@bbsbec.ac.in)

DOI: 10.7763/IJIEE.2013.V3.299

as pattern classifiers and as nonlinear adaptive filters. We will provide a brief overview of the theory, learning rules, and applications of the most important neural network models. Definitions and Style of Computation An Artificial Neural Network is an adaptive, most often nonlinear system that learns to perform a function (an input/output map) from data. Adaptive means that the system parameters are changed during operation, normally called the training phase. After the training phase the Artificial Neural Network parameters are fixed and the system is deployed to solve the problem at hand (the testing phase). The Artificial Neural Network is built with a systematic step-by-step procedure to optimize a performance criterion or to follow some implicit internal constraint, which is commonly referred to as the learning rule. The input/output training data are fundamental in neural network technology, because they convey the necessary information to "discover" the optimal operating point. The nonlinear nature of the neural network processing elements (PEs) provides the system with lots of flexibility to achieve practically any desired input/output map, i.e., some Artificial Neural Networks are universal mappers . There is a style in neural computation that is worth describing. [3]



The style of neural computation.

Fig. 1.1. General structure showing performance of the artificial neural network .

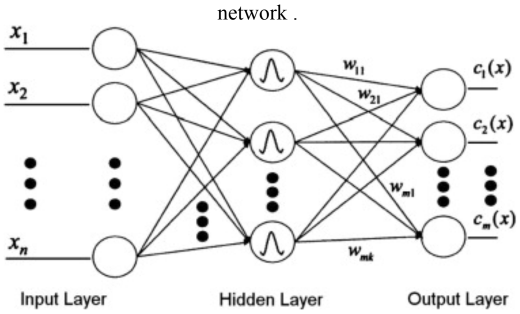


Fig. 1.2. General structure of the radial basis function neural network.

A. Radial Basis Function

In an MLP, the net input to the hidden layer is a linear combination of the inputs as specified by the weights. In a radial basis function (RBF) network (Wasserman 1993), as shown in Fig. 1. 2, the hidden neurons compute radial basis functions of the inputs, which are similar to kernel functions in kernel regression (Hardle 1990).The net input to the hidden layer is the distance from the input vector to the weight vector. The weight vectors are also called *centers*. The distance is usually computed in the Euclidean metric, although it is sometimes a weighted Euclidean distance or an inner product metric. There is usually a bandwidth sj associated with each hidden node, often called *sigma*. The activation function can be any of a variety of functions on the nonnegative real numbers with a maximum at zero, approaching zero at infinity. The outputs are computed as linear combinations of the hidden values with an identity activation function for comparison[4]. typical formulas for an MLP hidden neuron and an RBF neuron are as follows:

$$y(x) = \sum_{i=1}^{N} w_i \varnothing(\|x - c_i\, c_i\|)$$

So, output depends on distance from point ‘c’.

Another parameter used in only RBF is spread. Significant of spread is shown as:

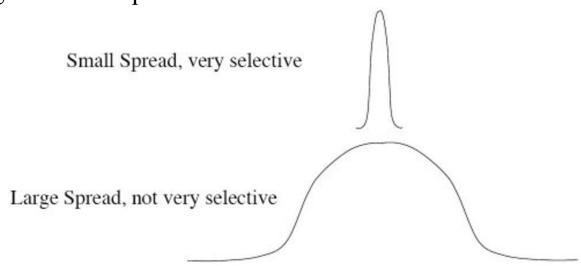


Fig. 1.3. Spread in the radial basis function neural network.

B. Digital Filter Using RBF

Since I use the radial basis function neural network for training the neural network. I need to train the RBFN for the different signals separately. For determining the RBF unit centers, we have used a ‘K-means’ clustering algorithm. The ‘K-means’clustering algorithm finds a set of clusters each with centers from the given training data. The cluster centers become the centers of the RBF units. the network is trained with different values of , , , , , as input parameters of low pass FIR filter. With similar ranges of input parameters I have designed a low pass FIR filter using Kaiser window which give actual filter coefficients. After this I gave any unknown value as input to trained network so that it give as value of h(n) i.e. impulse response of FIR filter or filter coefficients, this value of filter coefficients must be very to close to desired value of h(n).I then tried to reduce the error between desired and actual values of filter coefficients.

II. Kaiser Window

A desirable properly of the window function is that the function is if finite duration in the time domain and that the Fourier transform has maximum energy in the main lobe as a given back side lobe amplitude. Kaiser Window have had developed a simple approximation to these function in terms of zeroth order modified Bassel functions of the first kind.[5]

In Kaiser Window, the side lobe level can be controlled with respect to the main lobe peak by varying a parameter, α. The width of main lobe can be varied by adjusting the length of the filter [6]The Kaiser Window function is given by

$$W_F(n) = \begin{cases} \frac{I_o(\beta)}{I_o(\alpha)}, \text{for} |n| \le \frac{M-1}{2} \\ 0 \qquad \text{otherwise} \end{cases}$$

where, α is an important variable determined by Kaiser & the parameter β is expressed by

$$\beta = \alpha \left[1 - \left(\frac{2n}{M-1} \right)^2 \right]^{0.5}$$

The modified Bessel function of the first kind, $I_o(x)$, can be computed from power series expression given by

$$I_o(x) = 1 + \sum_{k=1}^{\infty}\left[\frac{1}{Ki}\left(\frac{x}{2}\right)^k\right]^2$$

III. Experimentation

The code has been implemented in MATLAB. The name Matlab stands for matrix laboratory MATLAB is an intercalate system whose basic data element is an array that does not require dimensioning. This allows solving many technical computing, in a fraction of the time for the simulation and the corresponding analyses of the given application the following framework and distributed design situations have been taken case of and then implemented feel typical set of settings.

Preparation of data sheet

With following that are:

1) Stop Band attenuation (SBT)
2) Transition width (TW)
3) Pass Band Ripple (PBR)
4) Sampling Frequency (SF)
5) Filter Length (N)

Filter coefficient are calculated and in this thesis works is carried out using approximately 30 such values of all the above parameters to calculate the filter coefficients. The range of different parameters has been taken which are:

SBT: (10.1040–9.3860)
TW: (50 –100)
PBR: (0.1_0.2)
N: (3–10)
PBF: (150_180)

Using this data set the Artificial Neural Network has been trained and can be use to calculate filter coefficients for input parameters in this range. Now, ANN is use to design the low pass FIR filter. There is very no difference in the Ann results and the calculated results.

IV. Results and Discussion

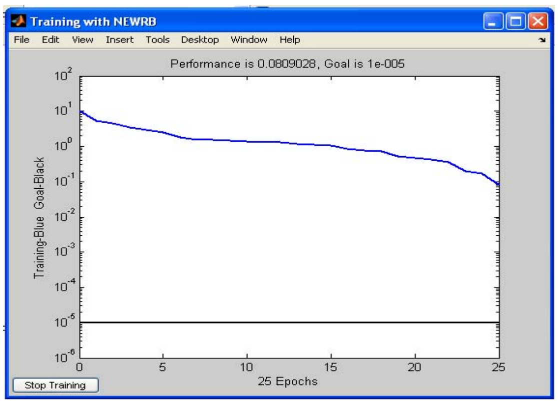


Fig. 2. Training of neural network

The data sheet (XLS format) is formed using Digital signal processing tool of Matlab Simulation. This data sheet in the matrix form of input parameter and target matrix form of target value has been trained. The Fig. 2 shows the results of trained network and the goal met criteria. Fig 3, III&IV shows the error graph b/w the ANN output and Kaiser Window output for different ranges of input parameters.

Error graph for the input values TW=50, SF=1.0, PBR=0.100, SBT= 10.1040 is shown below:

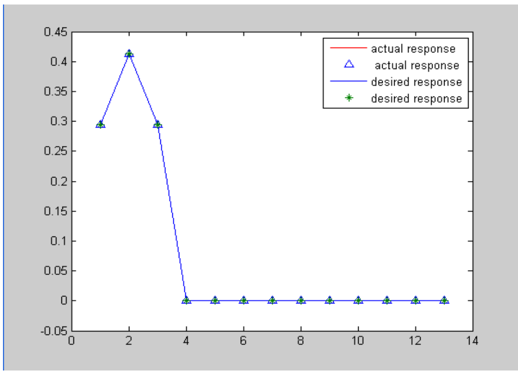


Fig. 3. Error graph

Error graph for input values TW=68, SF=3.2, PBR=0.132, SBT=10.6624is shown below:

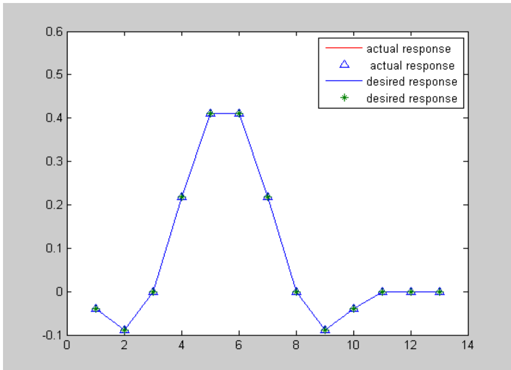


Fig. 4. Error graph

Error graph for input values TW=74, SF=4.4, PBR=0.142, SBT=8.6006 is shown

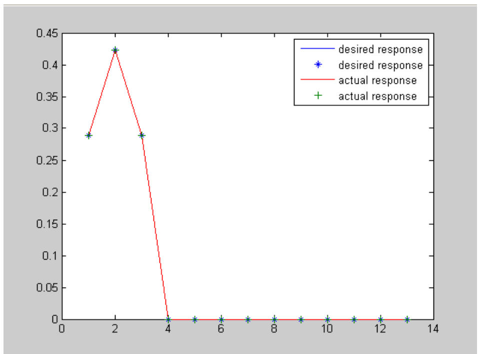


Fig. 5. Error graph

V. Conclusion

Artificial Neural Network is better and easy method of design of low Pass FIR Filter. Also, using Fourier series, Frequency sampling or Window methods the filter can be design but for each unknown parameter the filter coefficients have to calculated. In comparison with ANN, the trained network can calculate the filter Coefficient for unknown parameter in that specified range. Using ANN if error graphs are drawn between ANN output and Kaiser Window method is almost zero if we use radial basis function.

REFERENCES

[1] R. M. Garimella, "Finite Impulse Response FIR Filter Model of Synapses: Associated Neural Network," in *Proceeding of the Fourth Annual IEEE International Confidences on Natural Computation*, 2008, pp. 3304–3309.

[2] S. Haykins, *Neural Networks–A Comprehensive Foundation*, Prentice – Hall of India Private Limited, New Delhi, 2003.

[3] Z. Fan and P. Mars, "Access Flow Control Scheme for ATM Networks Using Neural-Network-Based Traffic Prediction," in *Proceeding of the IEEE*, vol. 144, October 1997, pp. 708–714,1997.

[4] A. Karami and M. S. Mohammadi, "Radial basis function neural network for power system load-flow," in *Proceeding of the Electrical Power and Energy Systems*, pp. 60–66, 2008.

[5] V. Aggarwal, O.W. Jin, and O. U. May, "Filter Approximation Using Explicit Time and Frequency Remain Specification," in *Proceeding of the Annual Symposium on Artificial Intelligence*, 2006, Seattle, Washington, 2006, pp. 174–165.

[6] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing – Principle, Algorithms and Applications*, Prentice –Hall of India, New Delhi, 2000.